Библиографические ссылки

1. Статистика уязвимостей Web-приложений за 2008 год [Электронный ресурс]. Режим доступа: http://ru.scribd.com/doc/21324267/WASC-Web-Application-Security-Statistics-2008-Russian

2. ModSecurity [Электронный ресурс]. Режим доступа: http://modsecurity.org

3. Barracuda Networks, Inc. (US) [Электронный ресурс]. Режим доступа: http://barracudanetworks.com

4. AdNovum Informatik AG [Электронный ресурс]. Режим доступа: http://adnovum.ch

5. A division of Virtual Graffiti, Inc. [Электронный ресурс]. Режим доступа: http://impervaguard.com

6. Компания «1С-Битрикс» [Электронный ресурс]. Режим доступа: http://1c-bitrix.ru

7. *Евтеев Д.* Методы обхода Web Application Firewall [Электронный ресурс]. Режим доступа: http://www.ptsecurity.ru/download/PT-devteev-CC-WAF.pdf

ВОЗМОЖНОСТИ ПРОГРАММНОГО УПРАВЛЕНИЯ СМАРТФОНОМ В ТЕХНОЛОГИЧЕСКИХ РЕЖИМАХ

В. В. Бакланов, И. А. Бильдинов
(Екатеринбург, УрФУ, banbmw@k66.ru)

По данным аналитической компании Strategy Analytics на конец 2-го квартала 2013 г. компания Google контролирует 67 % рынка операционных систем (ОС) планшетов. За год популярность операционной системы Android поднялась почти в 2 раза: с 18,5 млн поставленных устройств до 34,6 млн планшетных ПК, в то время как доля присутствия операционной системы Apple iOS за квартал упала с 47,2 % до 28,3 %. И это падение обусловлено популярностью Android [1]. Еще одна аналитическая компания J'son & Partners Consulting сообщает, что ОС Android лидирует на рынке ОС для смартфонов в России с долей в 77 %. Второе место, как в России, так и по миру, занимает iOS с долей 11 % по России и 13 % по миру [2].

Данная статистика свидетельствует о том, что большинство пользователей при покупке отдают предпочтение ОС Android на своих планшетных компьютерах и смартфонах. Безусловно, существует множество факторов выбора гаджетов с подобной ОС, но факт остается фактом – Android-устройства становятся все популярней.

Исходя из соображений популярности, у недобросовестных людей могут возникать корыстные идеи, направленные на сбор личной информации о пользователях, данных их аккаунтов, паролей и счетов банковских карт. Путем внедрения вредоносной программы, из-за невнимательности и беспечности пользователя данные идеи могут быть реализованы, что всегда приводит к неблагоприятным последствиям. Но манипуляции над Android-смартфоном можно производить не внедряя сторонние программы, а только используя команды, работая в технологическом режиме. Данные команды, следовательно, и их возможности не документированы, и в данной работе будут рассмотрены некоторые из них.

В качестве объекта исследований использовался смартфон Highscreen Explosion с версией ОС Android 4.0.3. Для его управления в технологическом режиме использовалась команда *service* с ее различными параметрами. Данная команда предназначена для работы с сервисами. *Сервис* – это компонент приложения, который позволяет приложению осуществлять те или иные длящиеся операции без взаимодействия с пользователем напрямую или позволяющий осуществлять взаимодействие с другими приложениями.

Справка команды *service* (*service -h*) имеет следующий вид:

Usage: service [*-h*|*-?*]
service list
service check SERVICE
service call SERVICE CODE [*i32 INT* | *s16 STR*] ...
Options:
i32: Write the integer INT into the send parcel.
s16: Write the UTF-16 string STR into the send parcel.

Таким образом, есть 3 варианта вызова данной команды:

– *service list* выводит список всех доступных сервисов. На рис. 1 представлен результат выполнения данной команды. Хочется заме-

тить, что список сервисов зависит от модели телефона. На модели, представленной в качестве объекта исследования, имеется 61 сервис, как и на смартфоне HTC Desire HD. На смартфоне же Samsung GALAXY S3 количество сервисов превышает 100 шт.;

– *service check SERVICE* проверяет наличие сервиса с именем *SERVICE*;

– *service call SERVICE CODE* [*i32 INT* | *s16 STR*] вызывает метод сервиса, где *SERVICE* – имя сервиса, *CODE* – порядковый номер метода сервиса;

– *i32 INT* и *s16 STR* – параметры в виде целого знакового числа и символьной UTF-16 строки соответственно.

Исходя из этого, встает следующая задача: если параметр *SERVICE* – это название сервиса из списка, то что является параметром *CODE* и какие числа и строки вводить вместо *INT* и *STR*. Для этого приведем пример команды и разберем ее синтаксис.

```
0       sip: [android.net.sip.ISipService]
1       phone: [com.android.internal.telephony.ITelephony]
2       iphonesubinfo: [com.android.internal.telephony.IPhoneSubInfo]
3       simphonebook: [com.android.internal.telephony.IIccPhoneBook]
4       isms: [com.android.internal.telephony.ISms]
5       samplingprofiler: []
6       diskstats: []
7       appwidget: [com.android.internal.appwidget.IAppWidgetService]
8       backup: [android.app.backup.IBackupManager]
9       uimode: [android.app.IUiModeManager]
10      usb: [android.hardware.usb.IUsbManager]
11      audio: [android.media.IAudioService]
12      wallpaper: [android.app.IWallpaperManager]
```

Рис. 1. Результат выполнения команды *service list*

Service call phone 2 s16 «123456789»

Данная команда производит дозвон на номер 123456789. При ее выполнении был получен следующий ответ (рис. 2).

Очевидно, что данные в ответе представлены в формате класса *Parcel*. Класс *android.os.Parcel* предназначен для упаковки сообщений и расширяет класс *java.lang.Object*.

Исходя из ответа посылки, ни текущий пользователь, ни текущий процесс не имеют права на выполнение звонка. В таком случае необходимо было получить права суперпользователя *root* и

```
Result: Parcel(
  0x00000000: ffffffff 00000049 0065004e 00740069 '....I...N.e.i.t.'
  0x00000010: 00650068 00200072 00730075 00720065 'h.e.r. .u.s.e.r.'
  0x00000020: 00310020 00300030 00390039 006e0020 ' .1.0.0.9.9. .n.'
  0x00000030: 0072006f 00630020 00720075 00650072 'o.r. .c.u.r.r.e.'
  0x00000040: 0074006e 00700020 006f0072 00650063 'n.t. .p.r.o.c.e.'
  0x00000050: 00730073 00680020 00730061 00610020 's.s. .h.a.s. .a.'
  0x00000060: 0064006e 006f0072 00640069 0070002e 'n.d.r.o.i.d...p.'
  0x00000070: 00720065 0069006d 00730073 006f0069 'e.r.m.i.s.s.i.o.'
  0x00000080: 002e006e 00410043 004c004c 0050005f 'n...C.A.L.L._.P.'
  0x00000090: 004f0048 0045004e 0000002e          'H.O.N.E.....    ')
```

Рис. 2. Результат выполнения команды
service call phone 2 s16 «123456789»

повторно вызвать команду с теми же параметрами. В результате дозвон по указанному номеру был успешно произведен. Таким образом, получив вышеупомянутые права и сделав дозвон на требуемый номер, можно подслушивать за владельцем телефона: с кем он ведет деловые переговоры, какие строит планы по развитию своей компании и где он будет сегодня вечером. Когда злоумышленником будет получена необходимая информация, он может завершить звонок командой

service call phone 5.

Данная команда выполняется без параметров *i32* и *s16* и завершает текущий вызов или входящий звонок. Организовать прослушивание можно, сделав дозвон на телефон жертвы, но предварительно включив в режим без звука:

service call audio 11 i32 0,

где *i32 0* – режим без звука; *i32 1* – режим вибрации; *i32 2* – режим со звуком. Также можно использовать многократно команду:

service call audio 1 i32 – 1 i32 1,

где *i32 – 1* – направление движение ползунка громкости. При *i32 –1* происходит уменьшение звука, при *i32 1* происходит увеличение звука, *i32 1* – флаг подтверждения изменения.

После того как отключен звук, принять входящий звонок можно командой

service call phone 6.

За перемещением смартфона, а в подавляющем большинстве и за его владельцем, можно наблюдать с помощью LAC (Local area code) и CID (Cell ID) координат. Вывести их можно следующей командой:

service call phone 27.

В результате будет получен объект Parcel с координатами. Используя сервисы в сети Интернет, можно узнать местоположение смартфона с точностью до нескольких десятков метров. Для этого требуется ввести MCC (Mobile country code, для Российской Федерации – 250), MNC (Mobile network code, у каждого сотового оператора свой код) и вышеупомянутые коды.

Обладая правами суперпользователя, мы также можем активировать модуль Wi-Fi:

service call wifi 13 i32 1.

После того как он был активирован, мы можем получить список сетей, к которым когда-либо подключался смартфон:

service call wifi 1.

Поскольку точки доступа публичных мест носят обычно похожее название с самим заведением, то можно понять, в каких публичных местах проводит свое время владелец смартфона.

Работая командами сервиса *bluetooth*, мы можем получить состояние данного модуля, т. е. активирован он или нет.

service call bluetooth 1.

Активировать его:

service call bluetooth 3.

Получить адрес:

service call bluetooth 5.

Получить имя или изменить его:

service call bluetooth 6,
service call bluetooth 7 s16 «DeviceName».

Запустить поиск bluetooth-устройств и сопряжение по указанному адресу:

service call bluetooth 13,
service call bluetooth 20 s16 «aa:bb:cc:dd:ee:ff».

Используя сервис *isms*, можно производить отправку сообщений на указанный номер с произвольным текстом:

service call isms 6 s16 «telNumber» i32 0 i32 0 s16 «smsText».

Хорошей особенностью для человека, использующего эту команду, является то, что она не появляется в отправленных сообщениях. Ее можно зарегистрировать только благодаря бдительности пользователя смартфона. Он может вовремя спохватиться по поводу утечки денежных средств, получив у провайдера запрошенную историю потребления трафика.

Осознав, какими возможностями обладает человек, имеющий доступ к технологическому режиму, становится ясно, что даже вышеописанных команд вполне достаточно, чтобы вести скрытое наблюдение за человеком, за его перемещениями или использовать его денежные средства. Также хочется сказать, что технологический режим не обязательно используется с корыстным умыслом, с помощью вариаций все той же команды *service* можно как получить информацию, например, о настройках мобильной связи, о параметрах SIM-карты, так и производить манипуляции с вибрацией, подсветкой сенсорных кнопок, блокировкой экрана.

Подводя итоги, можно смело заявить, благодаря данной команде и ее комбинациям удалось организовать управление смартфоном в технологическом режиме под управлением ОС Android версии 4.0.3. Такой контроль позволяет вести скрытую от владельца смартфона деятельность или же, наоборот, наносить ощутимый вред системе. Результаты такой деятельности будут намного продуктивней, если будут иметься конкретная цель и задачи, исходя из которых станут подбираться наиболее оптимальные способы и пути их решения.

Библиографические ссылки

1. Новости про операционные системы. 06.08.2013 [Электронный ресурс]. Режим доступа: http://nvworld.ru/news/android-controling-87-percent-tablet-market/

2. Digit : интернет-журнал о технологиях. 23.08.2013 [Электронный ресурс]. Режим доступа: http://digit.ru/technology/20130823/404655676.html

ИДЕНТИФИКАЦИЯ АВТОРА ТЕКСТА ПО СТОХАСТИЧЕСКИМ ХАРАКТЕРИСТИКАМ ПИСЬМЕННОЙ РЕЧИ

Н. Н. Бороденко
(Екатеринбург, УрФУ, natalia@gammaural.ru)

Вопрос об идентификации автора текста в глобальной сети стал одним из наиболее обсуждаемых представителями государственных органов и обществом.

Так, в Российской Федерации предложения о законодательном «запрете анонимности» в Интернете неоднократно высказывались руководителями правоохранительных ведомств в контексте борьбы с преступностью.

Правомерно пытаться найти ответы на следующие ключевые вопросы, связанные с идентификацией в Интернете:

1) можно ли создать универсальную, всеобщую, глобально признанную систему идентификации пользователей Интернета.

2) если создание такой системы возможно, то на каких принципах и с использованием каких технологий. Каковы могут быть цели такой идентификации. Как избежать в процессе ее создания и использования нарушений основных прав и свобод человека, в том числе на неприкосновенность частной жизни.

3) если создание системы, упомянутой в предыдущем пункте, невозможно, то по каким основным технологическим, организационным, правовым либо иным причинам.